Control Language for Distributed Clean

Zoltán Hernyák, Zoltán Horváth and Viktória Zsók

The present Clean-CORBA interface supports the distributed computing using functional programming paradigms. Our aim is to express computations in the form of distributed process-networks. Nowadays it is very prevailing developing and testing parallel applications on PC clusters [1]. Therefore it became important to provide such distributed environments, which makes possible the investigation of typically distributed applications with client-programs written in any programming language.

In our environment [2],[3] Clean functional client-programs can be interconnected via CORBA, making possible the distribution of the processes and the asynchronous communication. Skeletal programming in the functional language Clean [4] extensively can use the CORBA server objects referenced by the parameterized clients. However for a controlled behaviour of the process-network a strategy description is needed. The distributed evaluation of functions and the communication between clients needs high-level process description and control mechanism [5]. In order to define such behaviour strategies a control language and its semantics is required.

The present paper would like to define high-level language elements for the coordination of the component functions in distributed environment. Parallel compositional strategies and primitives are defined with direct control over the process-network. These elements can be used by programmer for writing applications without knowing the details of the distributed environment.

References

[1] Loidl, H.W., Klusik, U., Hammond, K., Loogen, R., Trinder, P.W.: GpH and Eden: Comparing Two Parallel Functional Languages on a Beowulf Cluster in: Gilmore, S. (ed.): Trends in Functional Programming, Vol. 2, pp. 39-52, Intellect, 2001.

[2] Hernyák Z., Horváth Z., Zsók V.: Clean-CORBA Interface Supporting Skeletons, to appear in: Proceedings of 6th International Conference on Applied Informatics, Eger, Hungary, January 27-31, 2004.

[3] Zsók V., Horváth Z., Varga Z.: Functional Programs on Clusters In: Striegnitz, Jörg; Davis, Kei (Eds.) (2003) Proceedings of the Workshop on Parallel / High-Performance Object-Oriented Scientific Computing (POOSC'03), Interner Bericht FZJ-ZAM-IB-2003-09, July 2003, pp. 93-100.

[4] Horváth Z., Zsók V., Serrarens, P., Plasmeijer, R.: Parallel Elementwise Processable Functions in Concurrent Clean, Mathematical and Computer Modelling 38, pp. 865-875, Pergamon, 2003.

[5] Pena, R, Rubio, F., Segura, C.: Deriving Non-Hierarchical Process Topologies, in: Hammond, K., Curtis, S.: Trends in Functional Programming, Vol 3. pp. 51-62, Intellect 2002.